Optimizing Sentiment Analysis on Amazon Food Product Reviews: Integrating Feature Extraction Methods with Classification Algorithms and Fine-tuned DistilBERT Models

Jiali Han
Khoury College of Computer Sciences, Northeastern University
han.jial@northeastern.edu

Abstract

Sentiment analysis of product reviews is a pivotal field within text mining and Natural Language Processing (NLP). Understanding the sentiments expressed in product reviews is not only vital for businesses to gauge customer satisfaction but also for other customers to make purchase decisions. This project evaluates the effectiveness of traditional sentiment analysis tools like VADER against advanced ensemble models that integrate multiple feature extraction techniques—such as BoW, TF-IDF, GloVe, and Word2Vec—with classification algorithms, including Logistic Regression, Multinomial Naive Bayes, and Support Vector Machines (SVM). An undersampling strategy was employed to address class imbalance. The aim is to develop a model that can surpass existing paradigms in accuracy and efficiency, providing brands with a nuanced understanding of consumer sentiments from online reviews. Future research will focus on refining the model's ability to discern between neutral, negative, and positive sentiments.

Keywords: Product Reviews, Sentiment Analysis, Machine Learning, Natural Language Processing, feature extraction, text classification

1. Introduction

The surge in sentiment analysis research is driven by the digital era's influence on consumer behavior and communication, particularly through e-commerce platforms and social media. These online reviews shape purchase decisions and are crucial for businesses aiming to enhance their offerings and stay competitive (Yaylı and Bayram, 2012). This project zeroes in on sentiment analysis of Amazon food product reviews.

Sentiment analysis, also known as opinion mining, is a natural language processing (NLP) technique that automates sentiment identification within text and then categorizes the sentiments as positive, negative, or neutral. In the domain of product reviews, this process is integral to extracting actionable insights from vast quantities of unstructured text data. This project leverages advanced machine learning algorithms and lexicon-based techniques to offer an in-depth sentiment analysis.

There are two approaches to sentiment analysis: the machine-learning approach and the lexicon-based approach (Anees et al., 2020). Fig 1. discusses the classification of sentiment analysis in more detail.

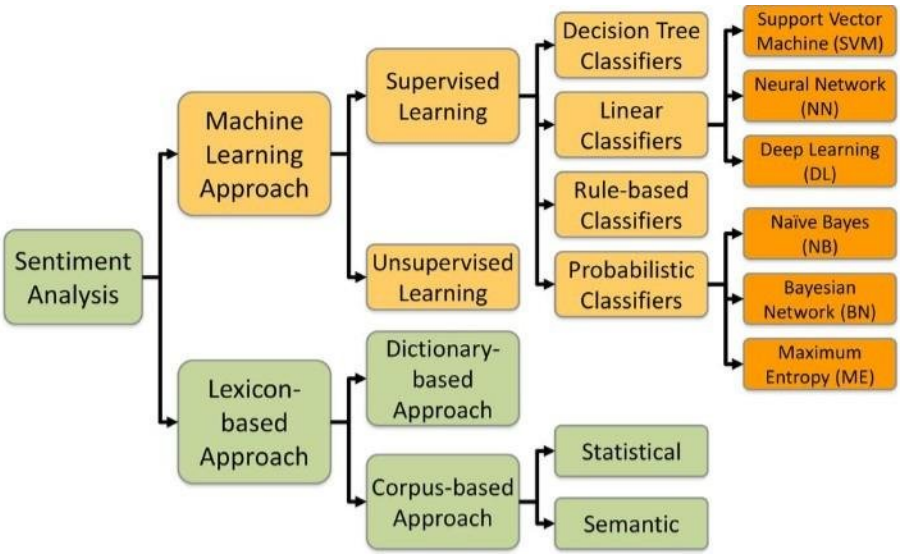


Figure 1: Classification of Sentiment analysis techniques

Lexicon-based sentiment analysis, which assigns predefined sentiment scores to words, is straightforward but often fails to capture the nuances and context of natural language, leading to inaccuracies. To address these shortcomings, researchers have adopted machine learning algorithms like Support Vector Machines (SVM), Naive Bayes, and Logistic Regression. These methods learn from labeled datasets to perform more context-aware sentiment analysis, but they can struggle with linguistic variations, unbalanced datasets, and the inherent semantic complexities of natural language.

Recent advancements in deep learning have led to the adoption of neural network-based models, particularly Recurrent Neural Networks (RNNs) and transformer-based architectures like BERT (Bidirectional Encoder Representations from Transformers). These models excel in understanding long-range dependencies and contextual information, significantly improving the accuracy and robustness of sentiment analysis.

Despite advancements in the field, sentiment analysis remains challenging due to the diversity of domains, languages, and cultural contexts. Moreover, maintaining a balance between accuracy and computational efficiency is essential for real-time applications. Therefore, this project aims to compare and refine models to address these challenges, enhancing precision and efficiency over existing methods.

2. Related Works

Sentiment analysis research has seen substantial growth, with notable studies spanning across various domains including Twitter, product reviews, and movie reviews.

In the study on datasets of movie reviews, Sharma and Dey (2012) explore the effectiveness of five feature selection methods and seven machine learning-based classification techniques. They found that the Gain Ratio was the most effective for sentiment feature selection, and SVM outperformed other classifiers for sentiment-based classification Similarly, O'Keefe and Koprinska (2009) evaluated a range of feature selectors and weights with Naive Bayes

and SVM classifiers, including novel methods, maintaining state-of-the-art classification accuracy while using a smaller feature set.

In Twitter sentiment analysis, Agarwal et al. (2011) introduced POS-specific prior polarity features, showing that tree kernel and feature-based models outperform the unigram baseline while Pandarachalil et al. (2015) presented an unsupervised method using sentiment lexicons, demonstrating good F1-score performance.

Regarding the sentiment analysis of product reviews, Xu Yun et al. (2015) from Stanford University applied supervised learning algorithms like perceptron, Naive Bayes, and SVM to Yelp's rating dataset, using various classifiers to determine precision and recall. Fang and Zhan (2015) conducted sentiment analysis on Amazon product reviews using Naive Bayes, concluding that Random Forest typically provided more accurate results. However, the downside is that the best F1 scores obtained from such experiments are fairly low, with values lower than 0.5. Because of the low accuracy, it is not recommended for decision-making. Elmurngi and Gherbi (2018) compared four supervised machine-learning algorithms for sentiment classification and found Logistic Regression to be the most accurate.

In the area of sentiment analysis, deep-learning neural networks have also been widely used. Shrestha and Nasoz (2019) used RNNs to train on product review vectors, developing a web service to predict ratings. Yang et al. (2020) proposed a model combining a sentiment lexicon with CNN (Convolutional Neural Network) and BiGRU (Bidirectional Gated Recurrent Unit), showcasing the synergy of sentiment lexicons and deep learning.

3. Methodology

This section outlines the methodology employed in this research, focusing on tasks and objectives, the scope and dataset, as well as the evaluation metrics used to assess the model performance.

3.1 Task and Objectives

The primary tasks of this project are to:

1) Evaluate the performance of VADER against advanced ensemble models that integrate multiple feature extraction techniques with classification algorithms in sentiment analysis of Amazon food product reviews.
2) Innovate beyond traditional approaches by fine-tuning a distilBERT model specifically tailored to the dataset to enhance accuracy and efficiency.
3) Compare the performance of the proposed model with both baseline and state-of-the-art sentiment analysis tools on a segmented dataset of product reviews.
4) Offer a comprehensive analysis of the results, highlighting the effectiveness and limitations of the proposed method.
5) Demonstrate the practical significance of this project on customer sentiment insights.

3.2 Scope and Dataset

The focus of this study is on sentiment analysis of English-language Amazon food product reviews. The dataset comprises over 500,000 reviews, spanning more than a decade up to October 2012. This dataset represents a wide range of sentiment intensity and class distribution, reflective of real-world data. The study excludes sentiment analysis of non-textual data and non-English reviews, as they fall outside the scope of this research. The goal is to develop a fine-tuned model that can accurately analyze and classify sentiments in Amazon food product review texts as positive, negative, or neutral.

3.3 Evaluation and Metrics

The performance of the models is evaluated using a classification report and a confusion matrix. The classification report includes precision, recall, f1-score, and support for each class, along with overall metrics such as accuracy and macro/weighted averages. The confusion matrix describes the performance of a classification model on a set of data with known true values. It allows me to visualize the accuracy of the model and highlight the correct and incorrect predictions broken down by each class, which is crucial for this imbalanced dataset as the "positive" class dominates the other two, skewing overall accuracy.

Given the imbalanced nature of this dataset, the macro-averaged F1-score is prioritized as the primary metric for model comparison. This ensures that each class contributes equally to the overall performance, which is essential in datasets where some classes are underrepresented (Fujino et al., 2008). This systematic evaluation approach ensures that each class contributes equally to the overall performance metric, which is crucial for datasets where some classes are underrepresented.

4. Experimental Work

This section presents the technical details of my project, including subsections that outline the step-by-step process of my experiments.

4.1 Data Collection and Preprocessing

The first step of sentiment analysis is the compilation of a diverse and representative dataset.

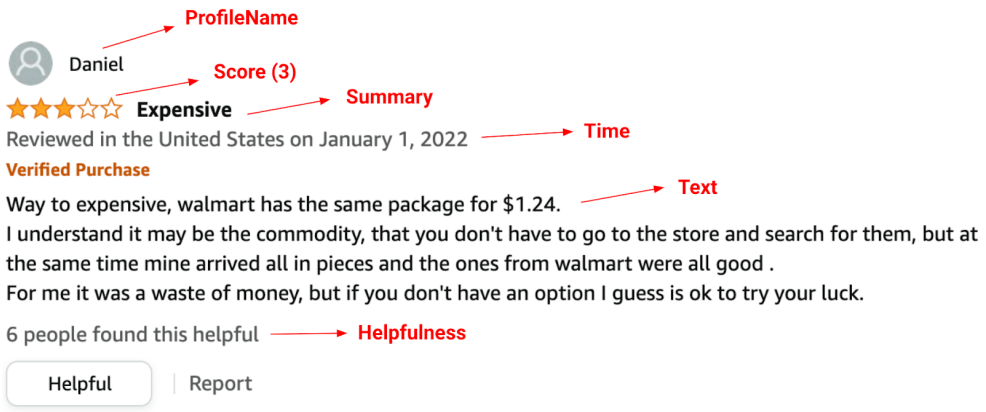


Figure 2: This is an example of an Amazon product review on Lady Fingers.

For this project, a dataset consisting of 568,454 Amazon food product reviews, available on Kaggle, was utilized. The dataset was further uploaded to Hugging Face for efficient access and management. An example of the dataset's structure and format can be seen in Figure 3.

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	Text
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	1	5	1303862400	Good Quality Dog Food	I have bought several of the Vitality canned d...
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	0	1	1346976000	Not as Advertised	Product arrived labeled as Jumbo Salted Peanut...
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	1	4	1219017600	"Delight" says it all	This is a confection that has been around a fe...
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	3	2	1307923200	Cough Medicine	If you are looking for the secret ingredient i...
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	0	5	1350777600	Great taffy	Great taffy at a great price. There was a wid...
...	...	...	...	...	...	...	...	...	...	...
568449	568450	B001EO7N10	A28KG5XORO54AY	Lettie D. Carter	0	0	5	1299628800	Will not do without	Great for sesame chicken..this is a good if no...
568450	568451	B003S1WTCU	A3I8AFVPEE8KI5	R. Sawyer	0	0	2	1331251200	disappointed	I'm disappointed with the flavor. The chocolat...
568451	568452	B004I613EE	A121AA1GQV751Z	pksd "pk_007"	2	2	5	1329782400	Perfect for our maltipoo	These stars are small, so you can give 10-15 o...
568452	568453	B004I613EE	A3IBEVCTXKNOH	Kathy A. Welch "katwel"	1	1	5	1331596800	Favorite Training and reward treat	These are the BEST treats for training and rew...
568453	568454	B001LR2CU2	A3LGQPJCZVL9UC	srfell17	0	0	5	1338422400	Great Honey	I am very satisfied ,product is as advertised,...

568454 rows × 10 columns

Figure 3: The dataset has 10 fields: Id, ProductId, UserId, ProfileName, HelpfulnessNumerator, HelpfulnessDenominator, Score, Time, Summary, and Text.

Initial preprocessing involved cleaning the data by removing duplicate reviews and extracting key columns (Score and Text) for a more focused analysis. This process reduced the dataset to 393,933 unique data points. To understand the dataset's composition, a distribution analysis of the scores was conducted, as shown in Fig 4. This analysis revealed an imbalance across the five classes (ratings 1 to 5), with a notably larger number of reviews in class 5.

Figure 4: The count of class 5 has 250,000 reviews, which is far more than the other 4 classes.

In the process of adapting the dataset's original 5-category score system to a 3-class sentiment classification framework, a pronounced class imbalance becomes apparent. This transformation involved categorizing scores above 3 as positive (1), below 3 as negative (-1), and equal to 3 as neutral (0). This reclassification scheme is illustrated in Fig 5.

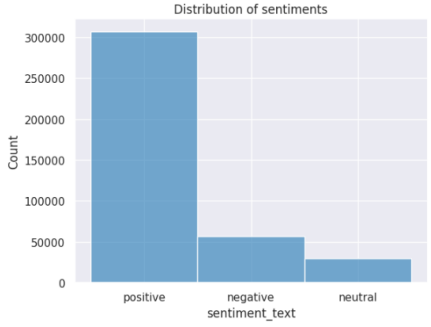


Figure 5: The count of "positive" has over 300,000 reviews, which is far more than the other 2 classes.

To explore key themes and terms in the reviews, word clouds were generated for each sentiment category as shown in Fig 6, 7, and 8. These word clouds revealed frequent terms in positive, neutral, and negative reviews, providing an insight into the prevalent vocabulary. Interestingly, the word "good" appeared across all sentiment categories, indicating its varied usage in different contexts. For example, "good" might be used in a sentence like "The product was not good", which is a context that might not be positive. Similarly, "good" in a neutral context ("The product was good, nothing special") doesn't convey strong positivity. In some cases, people also would use "good" in an ironic way in negative reviews such as "Good job on making the worst product I've ever used." In a word, this observation underscores the complexity of sentiment analysis, where a word's sentiment association can change based on its contextual use.

Figure 6: This is a word cloud generated from review texts classified as POSITIVE sentiment. Analysis of this word cloud reveals that the five most frequently occurring words in positively classified review texts are "love", "best", "good", "delicious", and "tasty".

Figure 7: This is a word cloud generated from review texts classified as NEUTRAL sentiment. Analysis of this word cloud reveals that the five most frequently occurring words in the neutrally classified review texts are "taste", "OK", "good", "great", and "okay".

Figure 8: This is a word cloud generated from review texts classified as NEGATIVE sentiment. Analysis of this word cloud reveals that the five most frequently occurring words in the negative classified review texts are “taste”, “good”, “flavor”, “product”, and “disappointed”.

Given the substantial size of the dataset, the labeled data is initially partitioned into a training set and a test set following an 80/20 ratio. Subsequently, to facilitate fine-tuning and evaluation of the model’s performance, 20% of the training set is segregated to form a validation set. Depending on machine learning algorithms or model architectures used, cleaned text data are further transformed into suitable numerical representations.

In the model exploration part, the performance of models with and without data resampling techniques was examined. As a popular way of dealing with imbalanced data, data resampling has two main methods: undersampling and oversampling in which samples are either reduced from the majority class or samples are added to the minority class (Shelke, 2017). According to Drummond and Holte (2003), under-sampling beats over-sampling by producing a reasonable sensitivity to changes in misclassification costs and class distribution. Therefore, in this project, data undersampling was employed, specifically undersampling of the majority class (positive sentiment).

4.2 Feature Extraction

Feature extraction is a crucial step in sentiment analysis, transforming textual data into a numerical format for processing by machine learning algorithms. This project compared various feature extraction techniques such as Bag of Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), word embeddings (GloVe and Word2Vec), and Bidirectional Encoder Representations from Transformers (BERT) embeddings.

4.2.1 BoW (Bag-of-Words)

The Bag-of-Words model, or BoW for short, represents a text document as if it were a “bag of words”, that is, an unordered set of words with their position ignored, keeping only their frequency in the document (Jurafsky and Martin, 2023). Implemented by the CountVectorizer tool in Python’s Scikit-learn library, it converts review text into a vector of term frequencies,

where each unique word in the corpus forms a feature. Package is available online[1].

4.2.2 TF-IDF (Term Frequency-Inverse Document Frequency)

Term Frequency-Inverse Document Frequency, or TF-IDF for short, measures a word's importance to a document in a corpus, considering both the word's frequency and its relative importance. Implemented by the TfidfVectorizer tool, it assigns higher weights to rare words in the corpus but frequent in individual documents, balancing the significance of common and uncommon terms. While offering an improvement over basic frequency counts, TF-IDF struggles to capture the semantic richness and domain-specific nuances. For example, in a customized dataset within a specialized domain, TF-IDF might underweight the words that are frequently used across multiple inputs but carry significant sentiment value in specific contexts, thereby diminishing their predictive value in sentiment analysis. Package is available online[2].

4.2.3 GloVe (Global Vectors for Word Representation)

Global Vectors for Word Representation, or GloVe for short, is a pre-trained word embedding method that maps words into a continuous vector space where semantically similar words are positioned closely. GloVe captures semantic relationships but assigns a fixed vector to each word, which may not capture context-specific meanings. Package is available online[3].

4.2.4 Word2Vec

Word2Vec, another pre-trained word embedding technique, uses neural networks to learn word associations from a large corpus of text. Word2Vec dense vector representations of words, capturing nuanced word relationships. Although powerful for many NLP tasks, it provides fixed embeddings and may not effectively capture words with multiple meanings based on context. Package is available online[4].

4.2.5 DistilBERT Embedding

DistilBERT, a lighter version of BERT, is a transformer-based model that converts text into sophisticated high-dimensional vectors (embeddings) (Sanh et al., 2019). These embeddings capture both individual words and their context within sentences, providing a deep understanding of language semantics and structure, a significant advancement over traditional feature extraction methods. Model is available on HuggingFace[5].

4.3 Sentiment Classification Instruments

This section delves into the sentiment classification instruments used in this project, detailing their implementation and role in the sentiment analysis process. Apart from traditional classifiers like Logistic Regression, Naive Bayes, and Support Vector Machines, this project

---

[1] github.com/scikit-learn/scikit-learn/blob/main/sklearn/feature_extraction
[2] github.com/scikit-learn/scikit-learn/blob/main/sklearn/feature_extraction
[3] github.com/stanfordnlp/GloVe
[4] code.google.com/archive/p/word2vec/
[5] huggingface.co/docs/transformers/model_doc/distilbert

also incorporates the fine-tuned DistilBERT model and VADER so this project can offer valuable insights due to their distinct approaches. DistilBERT, a streamlined variant of BERT, utilizes deep learning for accurate, context-rich text analysis, particularly effective after domain-specific fine-tuning. It excels in identifying subtle nuances in complex texts. Conversely, VADER, a lexicon-based model, is optimized for quickly assessing sentiments in short, informal texts, such as social media posts. By exploring and evaluating the performances of different classification instruments, this project aims to offer fresh insights into the effectiveness of these models on sentiment analysis on Amazon food products.

4.3.1 VADER (Valence Aware Dictionary and sEntiment Reasoner)

VADER, a lexicon-based sentiment analysis tool, excels in recognizing sentiment in social media texts. According to Elbagir and Yang (2019), VADER is particularly effective for short texts like tweets or reviews, where it can quickly and accurately assess sentiment without the need for extensive training or complex models. However, its reliance on a pre-defined lexicon can limit its adaptability to new or domain-specific jargon, which is frequently encountered in Amazon food product reviews. Code sample is available online[6].

4.3.2 Machine Learning Classifiers

In addition to VADER, various machine learning algorithms were utilized for sentiment classification, each with its unique strengths and limitations:

1) LR (Logistic Regression)

Logistic Regression, or LR for short, implemented by the LogisticRegression tool, is a linear model that is commonly used for classification. It is particularly effective for binary classification but can also handle multi-class problems. In the experiment, LG was applied with an LBFGS solver for optimization and was iterated up to 100 times to find optimal coefficients. Although a powerful tool, Logistic Regression may have limitations in handling highly complex and nonlinear relationships within data. Package is available online[7].

2) NB (Naive Bayes)

Naive Bayes, or NB for short, utilizing the MultinomialNB classifier, is a probabilistic algorithm known for its simplicity and efficiency in text classification. It assumes independence between predictors and scales well with large datasets. While generally effective, its assumption of feature independence can be a drawback in cases where the relationship between words is significant for sentiment analysis. Package is available online[8].

3) SVM (Support Vector Machines)

Support Vector Machines, or SVM for short, implemented with SGDClassifier with a hinge loss function, are particularly adept at handling high-dimensional spaces. This makes them

[6] vadersentiment.readthedocs.io
[7] github.com/scikit-learn/scikit-learn/blob/main/sklearn/linear_model/_logistic.py
[8] github.com/scikit-learn/scikit-learn/blob/main/sklearn/naive_bayes.py

suitable for text classification tasks, such as sentiment analysis. However, SVMs can require careful hyperparameter tuning and may not be as efficient with very large datasets or noisy data. Since Naive Bayes, Logistic Regression, and SVM are all considered conventional algorithms and are widely used in the field of sentimental analysis, these classifiers are used as benchmarks. Evaluation and results can be seen in Section 5. Package is available online[9].

4.3.3 Fine-tuned DistilBERT Model

To enhance the performance of the sentiment analysis model, this project also incorporated the fine-tuned DistilBERT models. As a distilled version of the original BERT model, DistilBERT retains 95% of its performance while having 40% fewer parameters and running 60% faster than bert-base-uncased (Sanh et al., 2019). It is trained to understand the context and semantics of text and is selected for this project due to its balance between performance and computational efficiency.

By dividing the dataset into 80% training and 20% validation sets, this project utilized a holdout validation method to fine-tune the DistilBERT model. To be specific, models are first trained on the training set and then validated on the validation set. To assess its generalization capability, the model is finally evaluated on a separate test set, consisting of data not seen during the training or validation phases. Key performance metrics for this evaluation include accuracy, precision, recall, and the F1-score.

The fine-tuning process of the DistilBERT model involved meticulous adjustments of various hyperparameters, including learning rate, batch size, and the number of training epochs. Given the computational demands and the time-intensive nature of training and evaluation, a primarily manual tuning approach was adopted in this project. For efficient experimentation, a smaller model variant, DistilBERT, was chosen, and initial trials were conducted on a subset of the dataset. This strategy facilitated rapid iteration over key hyperparameters, enabling effective fine-tuning while managing computational resources. Typical adjustments involved fine-tuning the learning rate, optimizing batch sizes, and determining the ideal number of training epochs for the model.

To mitigate the risk of overfitting, which is a common challenge in machine learning models, regularization techniques were employed. One such method was the implementation of weight decay, where a penalty is added to the loss function based on the magnitude of the model weights. This technique discourages the model from fitting too closely to the training data, thus promoting better generalization to unseen data. Additionally, early stopping was utilized as a strategy to prevent overtraining. By monitoring the model's performance on a validation set, the training process was halted once there was no further improvement in performance metrics. This approach not only helps in avoiding overfitting but also in optimizing the training duration, ensuring the model retains its ability to generalize effectively to new data.

[9] github.com/scikit-learn/scikit-learn/blob/main/doc/modules/sgd.rst

5. Results

This study utilized Python and Google Colab, along with supporting libraries, to execute data purification, visualization, pre-processing, and machine learning modeling. The sentiment classification models, created using supervised machine learning, involved building training, and testing sets from the review text dataset, which was divided into an 80/20 training/testing ratio. To fine-tune and assess model performance, 20% of the training data was set aside as a validation set. The performance of the models is evaluated by a classification report and a confusion matrix.

5.1 Model Performance

The performance metrics for various models, integrating multiple feature extraction methods with classification algorithms and augmented by undersampling, are presented in Figures 9 to 11. These figures illustrate the aggregate performance indicators for models applied to the dataset of product reviews. Additionally, Figure 12 compares the performance metrics of the VADER tool with and without the application of undersampling. Figures 13 and 14 focus on the fine-tuned DistilBERT model, showcasing its performance and the diverse hyperparameters employed during the fine-tuning process. Notably, the optimized DistilBERT model achieved a macro-averaged F1-score of 0.73 and an accuracy of 0.87, surpassing the benchmarks set by other models.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
BoW + LR	0.86	0.68	0.60	0.62
BoW + NB	0.83	0.62	0.61	0.62
BoW + SVM	0.86	0.70	0.57	0.58
BoW + LG + Undersampling	0.71	0.70	0.71	0.70
BoW + NB + Undersampling	0.70	0.71	0.70	0.70

Figure 9: This table presents a comparison of performance metrics across various sentiment analysis models employing the Bag of Words (BoW) feature extraction method. These models include combinations of BoW with Logistic Regression (LG), Naive Bayes (NB), and Support Vector Machine (SVM), with and without the application of random undersampling techniques. It is observed that while models with undersampling generally show a decrease in accuracy, they exhibit a notable improvement in precision, recall, and F1-score.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
TF-IDF + LR	0.86	0.70	0.59	0.61
TF-IDF + NB	0.79	0.56	0.36	0.34
TF-IDF + SVM	0.83	0.71	0.46	0.48
TF-IDF + LG + Undersampling	0.68	0.68	0.68	0.68

Figure 10: This table presents a comparison of performance metrics across various sentiment analysis models employing the Term Frequency-Inverse Document Frequency (TF-IDF) feature extraction method. These models include the combinations of TF-IDF with LG, NB, and SVM, with and without

using the random undersampling technique. It is observed that while the model with undersampling shows a decrease in accuracy, it exhibits a notable improvement in precision, recall, and F1-score.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
GloVE + LR	0.82	0.60	0.48	0.49
Word2Vec + LR	0.82	0.60	0.48	0.49
BERT embedding + LR	0.84	0.70	0.52	0.55

Figure 11: This table presents a comparison of performance metrics using 3 distinct word embedding methods combined a Logistic Regression classification algorithm. It is observed that the DistilBERT embedding model, among these three models, achieves a higher accuracy of and shows superior precision, recall, and F1-scores, suggesting that it provides a more effective representation for sentiment analysis in conjunction with logistic regression.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
VADER	0.73	0.55	0.47	0.47
VADER + undersampling	0.58	0.54	0.46	0.46

Figure 12: This table presents the performance metrics of the VADER with and without the application of undersampling. It is observed that when undersampling is applied to address the class imbalance, there is a notable decrease in accuracy, while precision, recall, and F1-score also change marginally.

Fine-tuned DistilBERT Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
1st model	0.78	0.26	0.33	0.29
2nd model	0.87	0.71	0.77	0.73

Figure 13: This table presents the performance comparison between two iterations of fine-tuned DistilBERT models. It is observed that there are significant enhancements in the 2nd model's ability to accurately classify sentiments, indicating a more effective fine-tuning process.

Parameters	1st training	2nd training
Learning rate	2e-4	5e-5
Batch size	8	8
Number of epochs	3	5
Weight decay	0.01	0.01
Seed	0	0
Optimizer	AdamW	AdamW

Figure 14: This table presents the hyperparameter configurations employed for the two iterations of fine-tuned DistilBERT models. It is observed that the learning rate of 2nd iteration was reduced to 5e-5 and the number of epochs was increased to 5, while the batch size and weight decay were maintained the same as the first iteration.

5.2 Analysis

For this study, the dataset comprised 568,454 reviews of food products sold on Amazon. The dataset underwent preprocessing, which included cleansing and tokenization. Subsequently, a

variety of feature extraction techniques and machine learning classifiers were applied to a training subset, constituting 80% of the data, amounting to 315,146 instances. Post-training, the models were validated, and a comprehensive assessment was carried out using a test set that included the remaining 20% of the data.

The results, as shown in Figure 15, reveal that the Bag of Words (BoW) approach, when integrated with Multinomial Naive Bayes and Logistic Regression classifiers, yielded more favorable results than the TF-IDF method and word embeddings. The outcome can be attributed to BoW's ability to effectively capture domain-specific terminology and the frequency of word occurrences in the relatively straightforward semantic structure of food product reviews. For example, BoW can better capture the frequent utilization of domain-specific terminology within food reviews (such as "delicious", "yummy", and "tasty"), while TF-IDF could potentially diminish the significance of these terms due to their commonality across multiple inputs. Furthermore, the conciseness of reviews may limit the effectiveness of TF-IDF's contextual weighting, whereas BoW's ability to encapsulate sufficient sentiment-indicative information remains unaffected. Finally, since food reviews usually are relatively short texts, the effectiveness of the TF-IDF method may be hindered, because it relies on a more contextual approach to weighting words. In contrast, the BoW method, which focuses on the presence of words rather than their context across documents, is not impacted by the length of the text and can still capture the necessary information to determine sentiment effectively.

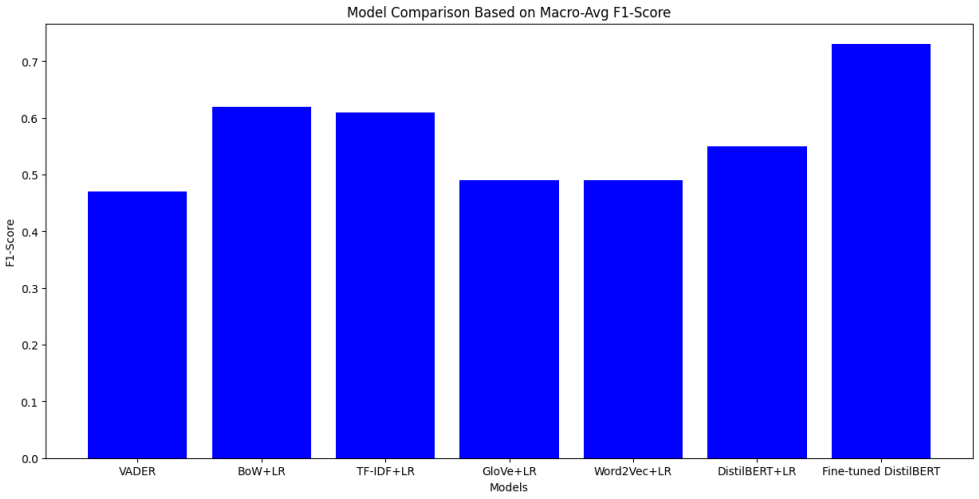


Figure 15: This diagram presents the performance metrics across various feature extraction methods employing the Logistic Regression Classifier. It is obvious that fine-tuned DistilBERT achieves the highest macro-averaged F1-score, following the Bag-of-Words and TF-IDF methods combined with the Logistic Regression Classifier.

Another thing worth noting is that the combination of the BoW method with the Logistic Regression classifier significantly outperforms VADER in sentiment analysis. The primary reason for this result could be the combined approach was trained directly on the specific dataset of Amazon food product reviews, which enables the model to understand the unique nuances and vocabulary, thereby enhancing its sentiment classification accuracy within this specific context.

Contrary to expectations, the BoW method outperformed advanced embedding techniques like GloVe, Word2Vec, and BERT in the sentiment analysis of this dataset. This result could be attributed to the following: First, the BoW model's high-dimensional yet sparse representation aligns well with sentiment analysis tasks involving large datasets by focusing on the presence or absence of sentiment-associated words. In the meantime, the straightforward language typical of food reviews, often replete with domain-specific terminology, is directly and effectively captured by BoW without the need for contextual nuance that embeddings attempt to provide. Embedding methods like GloVe and Word2Vec can sometimes introduce noise by mapping semantically similar but sentiment-different words close to each other. For instance, "good" and "bad" may appear in similar review texts and thus be closer in the embedding space, which could potentially lead to classifier confusion. Furthermore, the BoW approach can handle negations more effectively in sentiment analysis by treating them as separate features (e.g., "not good" vs. "good"), while word embeddings might lose this distinction as they focus on word similarity and context. Finally, pre-trained embeddings like GloVe, Word2Vec, or BERT are based on large, general-purpose corpora, which may not encapsulate the specific language nuances of food reviews. For example, certain words may have different sentiment valence in food reviews compared to general language use (e.g., "rich" might have a positive sentiment in the context of food but can be neutral or even negative in other contexts).

Though the combination of various feature extraction methods and different classification algorithms may give different results, it is observed that the performance of models greatly benefits from the implementation of an undersampling strategy. This is particularly evident in our experiments: models utilizing undersampling consistently outperform their counterparts that do not employ this strategy, as reflected in various performance metrics. For instance, a model integrating the Bag of Words (BoW) approach with Logistic Regression, further enhanced by undersampling, achieved a notable macro-average F1 score of 70%. In stark contrast, a similar model devoid of the undersampling strategy attained a significantly lower score of 62%. This difference underscores the critical role of undersampling in addressing data imbalance within our dataset as it can effectively mitigate the skewed distribution of classes, thus allowing for a more balanced representation that facilitates improved model learning and generalization. The application of this undersampling strategy before fine-tuning the DistilBERT models has also been shown to improve the accuracy and robustness.

However, an exception was noted with VADER, as shown in Figure 16, where undersampling slightly reduced its performance. As VADER's effectiveness is closely linked to its internal lexicon and rules, the reduction in dataset size and diversity through undersampling did not enhance its performance and instead limited the variety of expressions available for analysis. This distinction highlights the importance of considering the operational mechanisms of specific analytical tools when applying dataset manipulation techniques like undersampling.

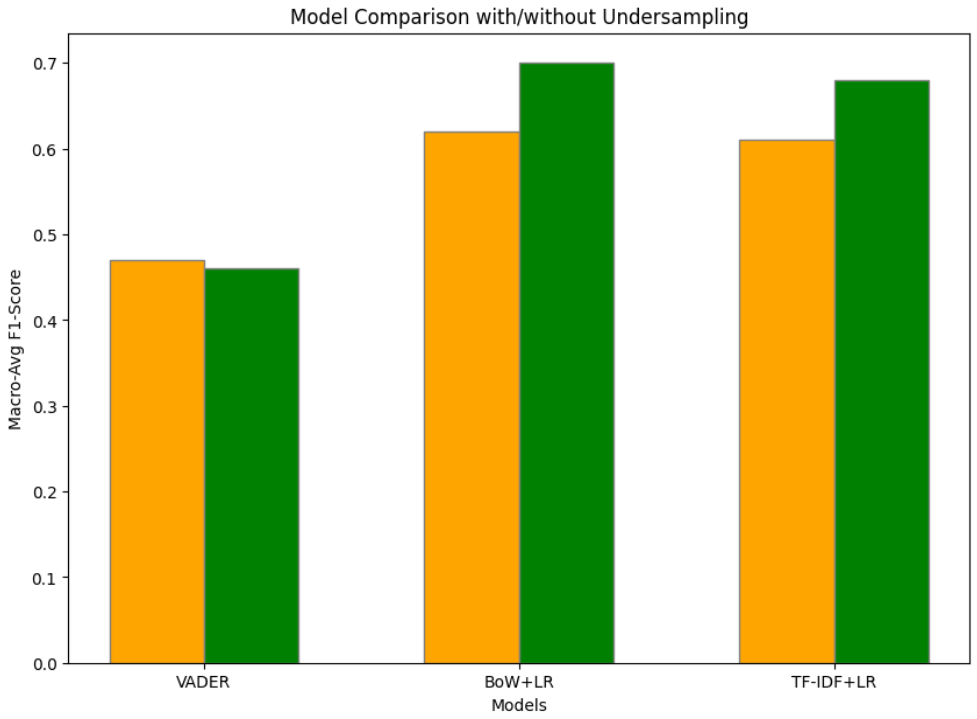


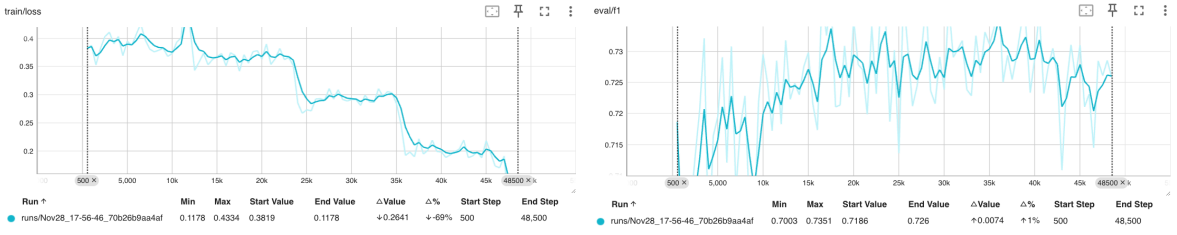
Figure 16: This diagram presents the performance metrics across the different combinations with and without using the undersampling technique. Most models using undersampling outperformed those that didn't. A striking example is our Bag of Words (BoW) approach combined with Logistic Regression. After using undersampling, this model achieved an impressive macro-average F1 score of 70%. Compare that to a similar model without undersampling, which only scored 62%.

In a word, the combination of the BoW approach and Logistic Regression offers a more flexible, data-driven, and adaptable framework for sentiment analysis in a domain-specific context like Amazon food product reviews, compared to other combinations and the predetermined lexicon and rule-based system like VADER. Also, it is obvious that not only the classification algorithm but also feature extraction has an important role in the process.

Due to the limited computational resources, a challenge for me as a student researcher, the fine-tuning of the DistilBERT model for sentiment analysis on Amazon product reviews was constrained to two training iterations. These iterations, each with its unique set of hyperparameters, incorporated an undersampling strategy to effectively address the issue of class imbalance.

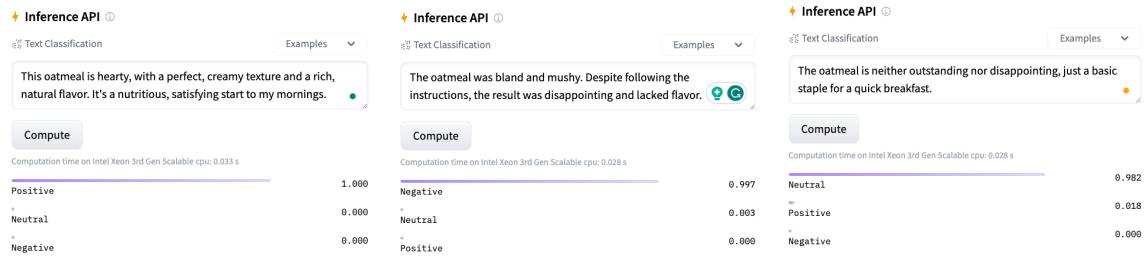
In the first training iteration, the model was configured with the following hyperparameters: a learning rate of 2e-4, training and evaluation batch sizes of 8, a seed value of 0, and the Adam optimizer with standard beta and epsilon values. A linear learning rate scheduler and an early stopping callback with the patience of five epochs were also employed. However, the outcomes were unsatisfactory, with the training loss stagnating around 1.0, validation loss around 0.7, and unchanged metrics including accuracy (0.78), precision (0.26), recall (0.33), and F1-score (0.29). These suboptimal results are attributed to a potentially high learning rate and early stopping may have prematurely halted training before the model adequately fitted the dataset.

To address these issues, the second iteration omitted early stopping and modified the learning rate to 5e-5 while extending training to 5 epochs. This led to significant improvements (as shown in Figures 17 and 18: training loss decreased to around 0.1, validation loss maintained at about 0.8, and marked enhancements in accuracy (0.87), precision (0.71), recall (0.77), and F1 score (0.73). The lower learning rate allowed for more precise weight adjustments, stabilizing the model's convergence. Additionally, the extended training duration facilitated deeper learning from the data patterns, culminating in better performance metrics.



Figures 17 & 18: These 2 images were screenshots from HuggingFace's Tensorboard of my fine-tuned model. They present the remarkable enhancement in metrics and underscore the delicate balance needed in tuning hyperparameters to refine the model's learning process and achieve better sentiment analysis results. For more metrics, please visit [HuggingFace's Tensorboard].

With the use of Inference API provided by Hugging Face, the finetuned model's performance on food product reviews can also be tested easily. Results can be shown in Figures 19 to 21.



Figures 19, 20, and 21: These 3 images were screenshots from HuggingFace's Inference API of my fine-tuned model. They present this model can determine with 100% certainty when analyzing a positive-sentiment review, can give a 99.7% possibility when analyzing a negative-sentiment review, and can predict with a 98.2% probability when analyzing a neutral-sentiment review. For more tests, please visit the [HuggingFace's Model Card].

In summary, the improvements in the second iteration underscore the importance of selecting appropriate hyperparameters, particularly a reduced learning rate and an increased number of epochs, tailored to the dataset's complexity. These adjustments allowed the model to learn from the dataset with greater efficacy, suggesting that the chosen values were better suited to the complexity and scale of the dataset. It is essential to note that optimal hyperparameter settings, particularly the learning rate, depend heavily on the dataset characteristics and the congruence between the source and target domains (Li et al., 2020). Therefore, future work should explore this interplay, aiming to refine hyperparameter optimization for enhanced model performance in sentiment analysis of Amazon food product reviews.

5. Conclusion

This project explores various sentiment analysis techniques applied to Amazon food product reviews. A key discovery is the better performance of the Bag of Words (BoW) approach combined with Logistic Regression over more complex feature extraction methods like TF-IDF, GloVe, Word2Vec, and even BERT embeddings. This finding challenges the notion that complexity invariably leads to improved performance, especially in datasets with specific linguistic characteristics.

The fine-tuning of the DistilBERT model, despite limited computational resources, further demonstrates the critical role of hyperparameter optimization in enhancing model efficacy. Adjustments in learning rate and training epochs notably improved the model's performance, highlighting the importance of tailored model configuration. The project also highlights that strategies like undersampling are not universally effective across all models, as evidenced by VADER's performance.

These findings open pathways for further research, especially in advanced optimization techniques. For example, deeper exploration into the impact of hyperparameters across diverse datasets, implementation of advanced cross-validation methods for optimization, and the potential of ensemble methods and deep learning in domain-specific sentiment analysis.

In essence, this research contributes valuable insights into sentiment analysis, particularly in the realm of online product reviews. It emphasizes the significance of methodical model optimization and the careful consideration of dataset characteristics, which are crucial for the development of effective sentiment analysis tools. These findings also hold practical implications, aiding businesses and researchers in the understanding and leveraging of consumer sentiments in the ever-evolving landscape of online product reviews.

6. Ethics

This research on sentiment analysis of Amazon food product reviews emphasized ethical considerations, particularly concerning data privacy, algorithmic bias, transparency, societal impact, and future ethical challenges. The dataset, consisting of publicly available Amazon reviews, was handled with strict adherence to data privacy norms. No personal or identifiable information from reviewers was used, ensuring the study respected individual privacy rights. The research also addressed the potential for algorithmic bias, a significant concern in sentiment analysis. Efforts were made to minimize bias, with algorithms rigorously tested and refined to prevent favoring or prejudicing any sentiment or group, thereby maintaining fairness. The methodologies and algorithms were also transparently detailed to allow replicability by other researchers, contributing to the scientific community's efforts to assess, validate, and build upon this work. The study recognized the influence of sentiment analysis tools on consumer behavior and business strategies, emphasizing the need for ethical use to enhance user experience and market research, rather than manipulating opinions or spreading misinformation. Looking ahead, the research advocates for maintaining these ethical standards in future works involving more advanced models and larger datasets.

7. Limitations

This project faced several limitations that impacted its broader applicability. First and foremost, computational constraints restricted the fine-tuning of the DistilBERT model to just two iterations with distinct hyperparameters, limiting the exploration of more advanced configurations and extensive hyperparameter tuning. Second, only focusing on Amazon food product reviews, this project may not capture the variety and complexity of other text datasets, thus limiting the generalizability of its findings to other domains. Third, though exploring various feature extraction methods and classification algorithms, the project did not encompass the entire spectrum of NLP methodologies. The undersampling strategy, while effective for most models, did not benefit all, like VADER, indicating the need for more tailored dataset manipulation techniques. Fourth, the process of hyperparameter selection for the DistilBERT model was largely manual and subject to my personal discretion, without the employment of automated methods like grid search due to computational limits. Future work should address these limitations by using cross-validation, regulation, and random-grid search methods to optimize model performance, thereby enhancing the robustness and scope of the sentiment analysis models. Last but not least, based on Dr. Church's feedback, it is noted that negative reviews are often lengthier and more detailed, indicating the need to include longer texts for a more balanced analysis. Additionally, considering the potential inauthenticity of positive reviews and the importance of addressing class imbalance, future work may include implementing oversampling methods to reduce bias in self-posted business reviews and to balance the predominance of positive reviews. It is also suggested that experimenting with different methods of running these models could lead to improved outcomes.

References

Yaylı, A., & Bayram, M. (2012). E-WOM: The effects of online consumer reviews on purchasing decisions. *International Journal of Internet Marketing and Advertising*, 7(1), 51-64.

Fang, X., & Zhan, J. (2015). Sentiment analysis using product review data. *Journal of Big Data*, 2(1), 1-14.

Anees, A. F., Shaikh, A., Shaikh, A., & Shaikh, S. (2020). Survey paper on sentiment analysis: Techniques and challenges. *EasyChair*, 2516-2314.

Xu, Y., Wu, X., & Wang, Q. (2015). Sentiment analysis of yelp's ratings based on text reviews. In *2015 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)* (Vol. 17, No. 1, pp. 117-120).

Sharma, A., & Dey, S. (2012). A comparative study of feature selection and machine learning techniques for sentiment analysis. In *Proceedings of the 2012 ACM research in applied computation symposium* (pp. 1-7).

O'Keefe, T., & Koprinska, I. (2009). Feature selection and weighting methods in sentiment analysis. In *Proceedings of the 14th Australasian document computing symposium, Sydney* (pp. 67-74).

Elmurngi, E. I., & Gherbi, A. (2018). Unfair reviews detection on amazon reviews using sentiment analysis with supervised learning techniques. *J. Comput. Sci.*, *14*(5), 714-726.

Shrestha, N., & Nasoz, F. (2019). Deep learning sentiment analysis of amazon. com reviews and ratings. *arXiv preprint arXiv:1904.04096*.

Yang, L., Li, Y., Wang, J., & Sherratt, R. S. (2020). Sentiment analysis for E-commerce product reviews in Chinese based on sentiment lexicon and deep learning. *IEEE access*, *8*, 23522-23530.

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. J. (2011). Sentiment analysis of twitter data. In *Proceedings of the workshop on language in social media (LSM 2011)* (pp. 30-38).

Pandarachalil, R., Sendhilkumar, S., & Mahalakshmi, G. S. (2015). Twitter sentiment analysis for large-scale data: an unsupervised approach. *Cognitive computation*, *7*(2), 254-262.

Fujino, A., Isozaki, H., & Suzuki, J. (2008). Multi-label text categorization with model combination based on f1-score maximization. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*.

Shelke, M. S., Deshmukh, P. R., & Shandilya, V. K. (2017). A review on imbalanced data handling using undersampling and oversampling technique. *Int. J. Recent Trends Eng. Res*, *3*(4), 444-449.

Drummond, C., & Holte, R. C. (2003). C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II* (Vol. 11, pp. 1-8).

Jurafsky, D., & Martin, J. H (2023). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*.

Elbagir, S., & Yang, J. (2019). Twitter sentiment analysis using natural language toolkit and VADER

sentiment. In *Proceedings of the international multiconference of engineers and computer scientists* (Vol. 122, p. 16).

Qaiser, S., & Ali, R. (2018). Text mining: use of TF-IDF to examine the relevance of words to documents. *International Journal of Computer Applications*, *181*(1), 25-29.

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Li, H., Chaudhari, P., Yang, H., Lam, M., Ravichandran, A., Bhotika, R., & Soatto, S. (2020). Rethinking the hyperparameters for fine-tuning. *arXiv preprint arXiv:2002.11770*.

J. McAuley and J. Leskovec. *From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews*. WWW, 2013.